

Lab 2: Practice Code

Probability Distributions, Statistical Inference, and Ordinary Least Squares

Your name

October 4, 2024

Prerequisite

```
# Good practice to remove all objects from the workspace
rm(list = ls())

# Use library() for packages you need, or source() for other R files.
library(tidyverse)

# Setting the seed ensures that we get the same random draw over and over again.
set.seed(20201009)

rnorm(5) # Check
```

```
## [1]  0.8315079 -0.9818884  1.1522644 -0.4687453 -0.8344489
```

Probability Distributions

0. Calculate the following operations by hand (... meaning by R)

a)

$$\sum_{i=1}^5 i =$$

b)

$$\prod_{i=1}^5 i =$$

c)

$$5! \times 10^{3!} \times e^4 =$$

```
# a)
```

```
# b)
```

```
# c)
```

1. Build a Bernoulli distribution using the `sample()` function, where the probability of *success* is 0.7. Run `?sample` if you are unsure how the function works.

```
# Create an imaginary person to flip the coin once for you
```

2. How do you know if it is working properly? Conduct simulation with for loop to check if the assigned probabilities are matched with the empirics

```
# Specify the number of simulations
sims <- 10000

# Specify the probability

# Create an empty vector as "container"
BernResult <- vector(mode = "numeric",
                     length = sims)

# For loop
for (i in 1:sims) {
  BernResult
}

mean(BernResult)
```

```
## [1] 0
```

3. Plot the above Bernoulli distribution

4. Based on the above, generate a binomial distribution, with number of trials equal to 10, without using `rbinom()`

```
# Create an imaginary person to flip the coin ten times for you
# Let's test it outside of the loop:

# Create number of simulations and an empty vector as container
BinoResult <- vector(mode = "numeric", length = sims)

for (i in 1:sims) {

  # Create an imaginary person to flip the coin ten times for you
  flips <- 0

  # Sum up the number of "success" for that person
  count <- 0
```

```
# Store it into the container; repeat 10,000
BinoResult <- 0

}
```

5. Plot the above binomial distribution

6. Explore the `rbinom`, `dbinom`, `pbinom` functions. What do they do? Answer the following questions:

- The probability of a coin landing on head is 0.7. If you were to flip the coin 10 times, what is the probability of getting exactly 7 heads?
- What is the probability of getting 7 heads or less?
- How do you know (b) is true?

```
# a) Pr(exactly 7 heads) -> PMF

# b) Pr(7 heads or less) -> CMF/CDF

# c) Double check
```

Review of Least Squares Estimation

In this section, we will use the built-in dataset *Salaries* from the `car` / `carData` package, which provides salary data for 397 randomly selected U.S. university faculty members.

Description: This dataset contains information on the 2008-09 nine-month academic salaries for Assistant Professors, Associate Professors, and Professors at a U.S. college. The data were collected as part of the college's ongoing efforts to monitor salary differences between male and female faculty members.

In this exercise, we will focus on the following three variables:

- salary** (Y): Outcome variable representing the nine-month salary in dollars.
- sex** (X_1): Binary indicator for the respondent's reported sex (Female or Male).
- yrs.since.phd** (X_2): Count variable indicating the number of years since the respondent received their PhD.

```
# clean environment
rm(list = ls())

# Load the data
data("Salaries", package = "carData")

# Prepare data

df <-
  Salaries |>
  # optional: turn it into a tibble
  as_tibble() |>
```

```
# re-scale salary
mutate(salary=salary/1000) |>
# select predictors for analysis
select(salary,sex,yrs.since.phd)
```

1. Manually calculate the mean salary for Male and Female faculty, and its difference. Then, use the `lm()` function to fit the following model: $salary = \alpha + \beta_1 * X_1 + \epsilon$. Check if the normality assumption of the errors hold.

```
# 1- differences in salaries between Male and Female

# 2- Regress salaries on sex using lm()

# 3- check the normality assumption
```

2. Now regress the following models:

$$\text{Model 2 : } Y_i = \alpha + \beta_2 X_{2i} + \epsilon$$

$$\text{Model 3 : } Y_i = \alpha + \beta_1 X_{1i} + \beta_2 X_{2i} + \epsilon$$

Recall that X_2 refer to `yrs.since.phd` and X_1 is `sex`.

```
# Regress salaries on years since phd using lm()
```

Regression coefficients can be thought of as comparisons across predicted values or as comparisons among averages in the data.

3. Investigate the residuals or predicted errors $\hat{\epsilon}$ from Model 3 in the previous exercise. What OLS assumption may be violated?

Hint: You can plot the residuals against each independent variable to identify potential issues. However, since we only have two variables and only one is non-dichotomous, you can use the `predict()` function to obtain the fitted values from model 3 and then plot the residuals against these fitted values. What do you observe?

```
# Plot the predicted residuals against the fitted values of the model
```

4. Devise a model specification that more accurately reflects the observed data-generating process (DGP) in the previous exercise. Please note that in this new model specification, you should not incorporate any additional predictors (e.g. rank or discipline) from the dataset.

```
## check residuals
```

5. Compare the models estimated with `lm()`. Which one provides the best fit? *Optional:* create a customized `function()` that computes the Mean Squared Error (MSE) of each model.

```
# compare the results
```

```
# Optional: define a function to calculate the Mean Squared Error
```